10-16-00

A

PTO/SB/05 (08-00)
Approved for use through 10/31/2002. OMB 0651-0032
U.S Patent and Trademark Office: U.S DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

Please type a plus sign (+) inside this box → +

UTILITY PATENT APPLICATION TRANSMITTAL *(Only for new nonprovisional applications under 37 CFR 1 53(b))*	Attorney Docket No.	004860.P2564
	First Inventor	Jerome R. Bellegarda
	Title	M & A FOR SPEECH RECOGNITION USING SEMANTIC INFERENCE AND WORD AGGLOMERATION
	Express Mail Label No.	EL034436293US

10/13/00

APPLICATION ELEMENTS

See MPEP chapter 600 concerning utility patent application contents

ADDRESS TO: Assistant Commissioner for Patents
Box Patent Application
Washington, DC 20231

1. ☒ Fee Transmittal Form (e.g., PTO/SB/17)
(Submit an original and a duplicate for fee processing)

2. ☐ Applicant claims small entity status. See 37 CFR 1.27.

3. ☒ Specification   [Total Pages **37**]
(preferred arrangement set forth below)
- Descriptive title of the Invention
- Cross References to Related Applications
- Statement Regarding Fed sponsored R & D
- Reference to sequence listing, a table, or a computer program listing appendix
- Background of the Invention
- Brief Summary of the Invention
- Brief Description of the Drawings (*if filed*)
- Detailed Description
- Claim(s)
- Abstract of the Disclosure

4. ☒ Drawing(s) *(35 U.S.C. 113)*   [Total Sheets 5]

5. Oath or Declaration   [Total Pages 5]
   a. ☐ Newly executed (original or copy)
   b. ☐ Copy from a prior application (37 C.F.R. § 1.63(d)) *(for continuation/divisional with Box 17 completed)*
      i. ☐ DELETION OF INVENTOR(S)
      Signed statement attached deleting inventor(s) named in the prior application, see 37 CFR 1 63(d)(2) and 1 33(b)

6. ☐ Application Data Sheet. See 37 CFR 1.76

7. ☐ CD-ROM or CD-R in duplicate, large table or Computer Program (*Appendix*)

8. Nucleotide and/or Amino Acid Sequence Submission *(if applicable, all necessary)*
   a. ☐ Computer Readable Form (CRF)
   b. Specification Sequence Listing on:
      i. ☐ CD-ROM or CD-R (2 copies); or
      ii. ☐ paper
   c. ☐ Statements verifying identity of above copies

ACCOMPANYING APPLICATION PARTS

9. ☐ Assignment Papers (cover sheet & document(s))
10. ☐ 37 C.F.R. § 3.73(b) Statement *(when there is an assignee)*   ☐ Power of Attorney
11. ☐ English Translation Document (*if applicable*)
12. ☐ Information Disclosure Statement (IDS)/PTO-1449   ☐ Copies of IDS Citations
13. ☐ Preliminary Amendment
14. ☒ Return Receipt Postcard (MPEP 503) *(Should be specifically itemized)*
15. ☐ Certified Copy of Priority Document(s) *(if foreign priority is claimed)*
16. ☐ Other: ....

17. If a CONTINUING APPLICATION, *check appropriate box, and supply the requisite information below and in a preliminary amendment:*
   ☐ Continuation   ☐ Divisional   ☐ Continuation-in-part (CIP)   of prior application No:

   Prior application Information: Examiner_____   Group/Art Unit: _____

For CONTINUATION OR DIVISIONAL APPS only: The entire disclosure of the prior application, from which an oath or declaration is supplied under Box 5b, is considered a part of the disclosure of the accompanying continuation or divisional application and is hereby incorporated by reference. The incorporation can only be relied upon when a portion has been inadvertently omitted from the submitted application parts.

18. CORRESPONDENCE ADDRESS

☒ Customer Number of Bar Code Label

08791

PATENT TRADEMARK OFFICE
(Insert Customer No or Attach bar code label here)

or ☐ Correspondence address below

Name	
Address	
City	State	Zip Code
Country	Telephone	Fax

Name (Print/Type)	Donna Jo Coningsby	Registration No. (Attorney/Agent)	41,684
Signature	*Donna Jo Coningsby*	Date	10/13/00

Burden Hour Statement. This form is estimated to take 0.2 hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, Washington, DC 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Assistant Commissioner for Patents, Box Patent Application, Washington, DC 20231.

FEE TRANSMITTAL
for FY 2000

Patent fees are subject to annual revision.

Complete if Known	
Application Number	
Filing Date	
First Named Inventor	Jerome R. Bellegarda
Examiner Name	
Group/Art Unit	
Attorney Docket No.	004860.P2564

TOTAL AMOUNT OF PAYMENT	($)	1,320.00

METHOD OF PAYMENT (check one)

1. ☒ The Commissioner is hereby authorized to charge indicated fees and credit any overpayments to.

Deposit Account Number: **02-2666**

Deposit Account Name: Blakely, Sokoloff, Taylor & Zafman LLP

☒ Charge Any Additional Fee(s) Required Under 37 CFR §§ 1.16, 1 17, 1 18 and 1.20

☐ Applicant claims small entity status. See 37 CFR 1 27.

2. ☒ Payment Enclosed:

☒ Check   ☐ Credit card   ☐ Money Order   ☐

FEE CALCULATION

1. BASIC FILING FEE

Large Entity		Small Entity		
Fee Code	Fee ($)	Fee Code	Fee ($)	Fee Description
101	710	201	355	Utility filing fee
106	320	206	160	Design filing fee
107	490	207	245	Plant filing fee
108	710	208	355	Reissue filing fee
114	150	214	75	Provisional filing fee

SUBTOTAL (1) ($) 710.00

2. EXTRA CLAIM FEES

	Extra Claims		Fee from below		
Total Claims	45	- 20 =	25	X 18.00 =	$450.00
Independent Claims	5	- 3 =	2	X 80.00 =	$160.00
Multiple Dependent				=	

**or number previously paid, if greater, For Reissues, see below

Large Entity		Small Entity		
Fee Code	Fee ($)	Fee Code	Fee ($)	Fee Description
103	18	203	9	Claims in excess of 20
102	80	202	40	Independent claims in excess of 3
104	260	204	135	Multiple Dependent claim, if not paid
109	80	209	40	**Reissue independent claims over original patent
110	18	210	9	**Reissue claims in excess of 20 and over original patent

SUBTOTAL (2) ($) 610.00

FEE CALCULATION (continued)

3. ADDITIONAL FEE

Large Entity		Small Entity			
Fee Code	Fee ($)	Fee Code	Fee ($)	Fee Description	Fee Paid
105	130	205	65	Surcharge - late filing fee or oath	
127	50	227	25	Surcharge - late provisional filing fee or cover sheet.	
139	130	139	130	Non-English specification	
147	2,520	147	2,520	For filing a request for reexamination	
112	920*	112	920	*Requesting publication of SIR prior to Examiner action	
113	1,840*	113	1,840	*Requesting publication of SIR after Examiner action	
115	110	215	55	Extension for response within first month	
116	390	216	195	Extension for response within second month	
117	890	217	445	Extension for response within third month	
118	1,390	218	695	Extension for response within fourth month	
128	1,890	228	945	Extension for response within fifth month	
119	310	219	155	Notice of Appeal	
120	300	220	155	Filing a brief in support of an appeal	
121	270	221	135	Request for oral hearing	
138	1,510	138	1510	Petition to institute a public use proceeding	
140	110	240	55	Petition to revive - unavoidable	
141	1,240	241	620	Petition to revive - unintentional	
142	1,240	242	620	Utility issue fee (or reissue)	
143	440	243	220	Design issue fee	
144	600	244	300	Plant issue fee	
122	130	122	130	Petitions to the Commissioner	
123	50	123	50	Petitions related to provisional applications	
126	240	126	240	Submission of Information Disclosure Stmt	
581	40	581	40	Recording each patent assignment per property (times number of properties)	
146	710	246	355	Filing a submission after final rejection (37 CFR 1.129(a))	
149	710	249	355	For each additional invention to be examined (37 CFR 1.129(b))	
179	710	126	355	Request for Continued Examination (RCE)	
169	900	169	900	Request for expedited examination of a design application	

Other fee (specify) _____
Other fee (specify) _____

* Reduced by Basic Filing Fee Paid   SUBTOTAL (3) ($)

SUBMITTED BY

Complete (if applicable)				
Name (Print/Type) **Donna Jo Coningsby**	Registration No. (Attorney/Agent)	41,684	Telephone	(503) 684-6200
Signature			Date	10/13/00

WARNING: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-2039.

UNITED STATES PATENT APPLICATION

for

METHOD AND APPARATUS FOR SPEECH RECOGNITION USING SEMANTIC INFERENCE AND WORD AGGLOMERATION

Inventor:

Jerome R. Bellegarda

Prepared by:

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN, L.L.P.
12400 Wilshire Boulevard
Seventh Floor
Los Angeles, California 90025
(503) 684-6200

METHOD AND APPARATUS FOR SPEECH RECOGNITION USING SEMANTIC INFERENCE AND WORD AGGLOMERATION

BACKGROUND OF THE INVENTION

Field of the Invention

5   The present invention relates generally to pattern recognition. More particularly, this invention relates to speech recognition systems that recognize commands using semantic inference and word agglomeration.

Copyright Notice/Permission

10   A portion of the disclosure of this patent document contains material that is subject to copyright protection. The copyright owner has no objection to the facsimile reproduction by anyone of the patent document or the patent disclosure as it appears in the Patent and Trademark Office patent file or records, but otherwise reserves all copyright rights whatsoever. The following notice applies to the software and data as

15   described below and in the drawings hereto: Copyright © 2000, Apple Computer, Inc., All Rights Reserved.

Background

As computer systems have evolved, the desire to use such systems for pattern recognition has grown. Typically, the goal of pattern recognition systems is to quickly

20   provide accurate recognition of input patterns. One type of pattern recognition system is a voice recognition system, which attempts to accurately identify a user's speech. Another type of pattern recognition is a handwriting recognition system. A speech recognizer discriminates among acoustically similar segments of speech to recognize

words, while a handwriting recognizer discriminates among strokes of a pen to recognize words.

An application of speech recognition is voice command and control (VCC), which enables a computer user to control a computer by voice rather than by using traditional user interfaces such as a keyboard and a mouse. Advances in speech recognition technology have enhanced the performance of VCC so that a computer can accurately perform a task by recognizing a command spoken within a restricted domain of vocabulary. However, existing VCC technology has limitations that diminish the usefulness of the technology to an average computer user.

Typical VCC applications employ a context-free grammar, such as a finite state grammar, that is a compact way of representing an exhaustive list of each and every command that the application can recognize. A finite state grammar is a particular implementation of a context-free grammar. These applications compare the spoken command to the list of commands underlying the context-free grammar. As a result, existing VCC applications that use a context-free grammar either reject or incorrectly recognize any utterance that is semantically accurate but syntactically out-of-grammar. This rigid framework requires the computer user to learn and memorize the specific commands that are compiled within the context-free grammar.

Semantic inference alleviates the problems associated with VCC applications that use a context-free grammar. Semantic inference is a more tolerant approach to language modeling that enables a computer to recognize commands that are out-of-grammar but semantically accurate, thereby allowing computer users to say what they mean rather than requiring them to speak from a pre-defined list of commands. For example, semantic

inference will enable a user to say "make a new spreadsheet" when the pre-defined wording of the command is "Open Microsoft Excel."

VCC applications that use semantic inference typically employ a speech recognition unit to provide a transcription of the user's spoken command. A semantic classification engine applies semantic inference to the transcription to determine the desired action. Some VCC applications using semantic inference replace the context-free grammar in a speech recognition unit with a statistical language model such as an n-gram. A statistical language model makes it possible for the speech recognition unit to transcribe, with a reasonably low error rate, whatever formulation the computer user chooses for expressing a command. This substitution prevents the speech recognition unit from rejecting out-of-grammar voice inputs before the semantic classification engine has the opportunity to evaluate the voice input for semantic similarity.

However, regardless of whether the VCC application uses a context-free grammar or a statistical language model, current implementations of semantic inference operate only at the word level. This is because the latent semantic analysis that comprises the process of semantic inference is an instance of the so-called "bag-of-words" model, which pays no attention to the order of words in the command. As a result, commands containing the same words in a different order are erroneously mapped to the same representation. For example, the commands "Change icons to list" and "Change list to icons" are indistinguishable, even though the underlying commands are very different. Thus, while latent semantic analysis is well-suited to capture large-span (i.e., semantic) relationships between words, it is inherently unable to capitalize on the local (i.e., syntactic or pragmatic) constraints present in the language. To avoid the erroneous

mapping of these types of commands to the same representation requires additional processing, such as performing back-off for sense disambiguation. The additional processing is undesirable since it consumes additional central processor unit (CPU) cycles and degrades performance. What is needed, therefore, is an improved method and apparatus for using semantic inference in a speech recognition system to more accurately recognize a command.

SUMMARY OF THE INVENTION

A method and apparatus for command recognition using semantic inference and word agglomeration is described herein. According to one aspect of the present invention, a method for recognizing a voice command comprises recognizing a sequence of words received as the voice command. The sequence of words is further agglomerated into a sequence of word n-tuples. Semantic inference is applied to the sequence of word n-tuples to recognize the voice command.

According to one aspect of the present invention, a machine-readable medium has stored thereon a plurality of instructions that, when executed by a processor, cause the processor to recognize a voice command by recognizing a sequence of words received as the voice command, to agglomerate the sequence of words into a sequence of word n-tuples, and to apply semantic inference to the sequence of word n-tuples to recognize the voice command.

According to one aspect of the present invention, an apparatus for recognizing a voice command includes a speech recognizer, a word agglomerator, and a semantic classifier. The speech recognizer recognizes a sequence of words received as the voice command, the word agglomerator agglomerates the sequence of words into a sequence of word n-tuples, and the semantic classifier applies semantic inference to the agglomerated word n-tuples to recognize the voice command.

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention is illustrated by way of example and not limitation in the figures of the accompanying drawings, in which like references indicate similar elements and in which:

Figure 1 is a block diagram illustrating a voice command recognition system according to one embodiment of the present invention;

Figure 2 is a flowchart illustrating the process followed in recognizing voice commands according to one embodiment of the present invention;

Figure 3 is a flowchart illustrating the process of training the command recognition system according to one embodiment of the present invention;

Figure 4 is a flowchart illustrating the process followed in classifying voice input according to one embodiment of the present invention;

Figure 5 illustrates one embodiment of a computing device suitable for use with one embodiment the present invention.

DETAILED DESCRIPTION

In the following description, various aspects of the present invention will be described. However, it will be understood by those skilled in the art that the present invention may be practiced with only some or all aspects of the present invention. For

5   purposes of explanation, specific numbers, materials and configurations are set forth in order to provide a thorough understanding of the present invention. However, it will also be apparent to those skilled in the art that the present invention may be practiced without these specific details.

Parts of the description will be presented in terms of operations performed by a

10   computer system, using terms such as data, flags, bits, values, characters, strings, numbers and the like, consistent with the manner commonly employed by those skilled in the art to convey the substance of their work to others skilled in the art. As is well understood by those skilled in the art, these quantities take the form of electrical, magnetic, or optical signals capable of being stored, transferred, combined, and otherwise

15   manipulated through mechanical and electrical components of the computer system; and the term computer system includes general purpose as well as special purpose data processing machines, systems, and the like, that are standalone, adjunct or embedded.

Additionally, various operations will be described as multiple discrete steps in turn in a manner that is helpful in understanding the present invention. However, the

20   order of description should not be construed as to imply that these operations are necessarily order dependent, in particular, the order of their presentations.

The present invention provides a method and apparatus for command recognition using semantic inference and word agglomeration. Semantic inference is a process that

uses latent semantic analysis (LSA) to capture the global semantic relationships between the words of a command by constructing a semantic space that characterizes co-occurrences between words and documents. LSA is a modification of a paradigm that was first formulated in the context of information retrieval, and it constructs the semantic

5   space by using a singular value decomposition of a word-document correlation matrix. A document refers to the instances of sentences, phrases, or other word groupings associated with a particular command. Depending on the command, the order of the words may be important in distinguishing one command from another. However, semantic inference, by itself, does not address this problem since it does not take word

10   order into account. Therefore, the method of the present invention proposes an extension to the basic LSA framework by embedding word agglomeration so that word order is taken into account in the construction of the semantic space. The word agglomeration/LSA extension captures the local syntactic constraints present in the commands by characterizing co-occurrences between word n-tuples and n-tuple

15   documents, where each word n-tuple is the agglomeration of n successive words, and each n-tuple document corresponds to each original document written in terms of all the word n-tuples it contains.

According to one embodiment, an audio input is initially provided to a speech recognizer that uses a context-free grammar or a statistical language model, or any other

20   technique that is capable of recognizing the words in the input. The recognized words are further agglomerated into a sequence of word n-tuples. The sequence of word n-tuples are then provided to a semantic classifier that, in conjunction with semantic anchors

created using training data, classifies the input as corresponding to a particular command. The appropriate action, based on the particular command, can then be generated.

Figure 1 is a block diagram illustrating speech recognition in a VCC system 100 using semantic inference and word agglomeration according to the present invention. A speech recognition unit 104 receives an audio input 102 and, using acoustic models 106 and a language model 108, generates a sequence of words 110. The audio input 102 is audio data that is input to the VCC system 100 and is intended to represent any type of audio data. Typically, the audio input 102 is a digitized representation of a human voice.

According to one embodiment of the present invention, the acoustic models 106 are hidden Markov models. Alternate embodiments can use different types of acoustic models, and any of a variety of conventional acoustic models other than hidden Markov models can be used. According to one embodiment of the present invention, the language model 108 is a context-free grammar, such as a finite state grammar, that is a compact way of representing an exhaustive list of each and every command that the VCC system 100 can recognize. Alternate embodiments of VCC system 100 can use different types of language models, including a conventional n-gram statistical language model (such as a bigram model, where $n=2$), where the probability of every word depends only on the n previous words. Hidden Markov models, n-gram language models, and context-free grammars are well-known to those skilled in the art and thus will not be discussed further except as they pertain to the present invention.

A word agglomeration unit 105 receives the sequence of words 110 and agglomerates them into a sequence of word n-tuples 111. According to one embodiment of the present invention, the process of word agglomeration performed by a word

agglomerator 105 is accomplished by replacing the sequence of words 110 in the command with the associated word n-tuple sequence, i.e., the sequence of all strings of n consecutive words present in the original word sequence, resulting in a sequence of word n-tuples 111. A semantic classification unit 112 receives the sequence of word n-tuples

5   111 and determines the correlation between the sequence of word n-tuples 111 and one or more semantic anchors 114. The correlation is the distance between a vector corresponding to the sequence of word n-tuples 111 and the vectors corresponding to the semantic anchors 114. The semantic classification unit 112 classifies the sequence of word n-tuples 111 as corresponding to the semantic anchor 114 with the closest

10   correlation. The semantic classification unit 112 sends a semantic representation 116 of the classified sequence of word n-tuples 111 to an action generation unit 118. The action generation unit 118 receives the semantic representation 116 and generates an action 120.

According to one embodiment of the present invention, the VCC system 100 separates different sequences of word n-tuples 111 from one another based on detected

15   pauses between words in the voice input 102. In this embodiment, if the VCC system 100 detects silence for greater than a threshold period of time, then the VCC system 100 interprets the silence as a pause between separate sequences of word n-tuples 111 corresponding to separate commands. The semantic classification unit 112 generates a separate semantic representation 116 for each separate sequence of word n-tuples 111.

20   Typical values for this threshold period of time range from approximately 300 milliseconds to approximately 3 seconds. In the illustrated implementation, the speech recognition unit 104 detects the pauses and indicates to the semantic classification unit 112 which sequences of word n-tuples 111 correspond to separate commands (e.g., by

sending a signal to the semantic classification unit 112, or by grouping words of a sequence of word *n*-tuples 111 together, e.g., by using a linked list). Alternatively, other methods can be used to identify the beginning and/or ending of a sequence of word *n*-tuples 111, such as a manual input from a user (e.g., selection of a particular box via a

5   graphical user interface and a pointer).

The action generation unit 118 uses the semantic representation 116 to determine the action 120 that is to be taken in response to the audio input 102 and generate the action 120. According to one embodiment, the action generation unit 118 maintains a mapping of possible semantic representations 116 to actions 120 (e.g., system commands

10   to display time, open files, close applications, etc.). It is to be appreciated that a wide range of commands can be supported by the VCC system 100, with the exact nature of the commands being dependent on the environment where the VCC system 100 is used. The action generation unit 118 is well known to those skilled in the art, and thus will not be discussed further except as it pertains to the present invention.

15   Figure 2 is a flowchart illustrating the process followed in performing speech recognition in a VCC system according to one embodiment of the present invention. Initially, the VCC system is trained in step 202, producing semantic anchors that are used to classify audio input later during recognition (in step 210, below). This training can be done by the manufacturer (e.g., entering training data when the system is built) and/or by

20   the user (e.g., the user speaking in the training data). In the illustrated embodiment, once the VCC system is initially trained, no further training is necessary for command recognition. However, a user may optionally perform additional training to, for example, customize a VCC system to a particular user's style.

After the VCC system is trained, an audio input 102 is eventually received in step 204. The speech recognition unit 104 recognizes and transcribes the audio input 102 into a sequence of words (i.e., the words of the input command) 110 in step 206. The word agglomeration unit 105 agglomerates the sequence of words 110 into a sequence of word n-tuples 111 in step 208. The semantic classification unit 112 classifies the sequence of word n-tuples 111 in step 210. The process of semantic classification generates a semantic representation 116 of the sequence of word n-tuples 111, and identifies, based on the semantic anchors produced during training in step 202, which command the semantic representation 116 actually represents. The semantic representation 116 is then used by the action generation unit 118 to carry out the desired action in step 212. As discussed in more detail below, the process of classification performed by the semantic classification unit 112 is based on the actual words that are part of the audio input 102 as well as the order of those words.

Figure 3 is a flowchart illustrating the process of training the command recognition system according to one embodiment of the present invention. According to one implementation, Figure 3 illustrates the process of step 202 of Figure 2 in more detail. Training data is entered into the system and mapped to a vector space having a particular dimension. For ease of explanation, in the discussions to follow the training data is referred to as sentences. However, it is to be appreciated that the training data can be sentences, phrases, or other word groupings. As noted earlier, instances of such units associated with a particular command form what is referred to as a "document" pertaining to that command. In the illustrated embodiment, the mapping is performed using an extension of the LSA framework that relies on a singular value decomposition of a word

n-tuple/n-tuple-document correlation matrix, where each word n-tuple is the agglomeration of n successive words as formed by the previously described process of word agglomeration, and each n-tuple document corresponds to each original document written in terms of all the word n-tuples it contains.

5   The word n-tuple/n-tuple-document correlation matrix captures the co-occurrences between word n-tuples and n-tuple documents by tabulating the number of times each word n-tuple occurs in each n-tuple document. Assuming that the training database includes N n-tuple documents and a total of M different word n-tuples, and assuming that the word n-tuples form the rows of the matrix and the n-tuple documents form the columns, then each element (i, j) of the word n-tuple/n-tuple-document

10   correlation matrix indicates how many times the corresponding word n-tuple M_i occurs in the n-tuple document N_j. Each n-tuple document from a column of the matrix can be represented as a single vector containing M values, with the resulting vectors defining a mapping between the vocabulary of word n-tuples and a vector space of dimension M.

15   As discussed in more detail below, this mapping can be refined by performing a singular value decomposition (SVD) of the correlation matrix, in which case the vector space is reduced to a dimension R, where $R < M$.

Note that word agglomeration does not impact the document dimension of the correlation matrix, but it does increase the word dimension by a factor theoretically up to

20   M^{n-1}, where M is the size of the underlying vocabulary. As a practical matter, a reasonable limit on the size of the word agglomeration is $n \leq 3$, which is compatible with typical VCC applications where M is on the order of a couple of thousand. For large vocabulary language modeling applications, however, M could reach up to 100,000.

Nevertheless, even large vocabulary language modeling applications can take advantage of word agglomeration as long as dimensionality is reduced before taking the SVD of the correlation matrix. One method for reducing dimensionality is to perform, for example, a distance-preserving random projection onto a subspace of double the expected SVD

5   dimension. The methods for reducing dimensionality are well known to those skilled in the art, and thus will not be discussed further except as it pertains to the present invention.

As illustrated in Figure 3, one or more sequences of words 110 (e.g. sentences) for each command document are first received in step 302. A "command document" refers to

10   a collection of one or more sentences, phrases, word grouping, or other sequences of words 110 all intended to identify the same command. The sequence of words 110 comprising the command document are each agglomerated in step 304 into a corresponding sequence of word n-tuples 111, which taken together result in the formation of a command n-tuple document.

15   Specifically, let $\mathcal{T}$ represent a training text corpus, i.e., a collection of N documents of interest, and let V represent the underlying vocabulary, i.e., the set of all the words appearing in $\mathcal{T}$, where $|V| = M$. To agglomerate the words in $\mathcal{T}$ the word agglomeration unit 105 first replaces each of the sequence of words in $\mathcal{T}$ by the associated word n-tuple sequence, i.e., the sequence of all strings of n consecutive words

20   present in the original word sequence. Note that this replacement, or mapping, of words to word n-tuples defines the associated n-tuple vocabulary $\overline{V}$ comprising all the word n-tuples appearing in $\mathcal{T}$, where the size of $\overline{V}$ is denoted by $\overline{M}$.

The process of word agglomeration in step 304 is further illustrated in the following example. Consider the following two commands:

$$\text{change pop-up to window} \tag{1}$$

and

$$\text{change window to pop-up} \tag{2}$$

Using the original LSA framework, these two commands are mapped onto the exact same point in the LSA semantic space, making them impossible to disambiguate. By embedding word agglomeration, however, the two commands are mapped onto different points in the LSA extension semantic space. The word agglomeration is accomplished by replacing the sequence of words in the command with the associated word n-tuple sequence, i.e., the sequence of all strings of n consecutive words present in the original word sequence. To illustrate, in the simple case $n = 2$, the commands (1) and (2) now become, respectively:

$$\text{\#:change change:pop-up pop-up:to to:window window:\#} \tag{3}$$

and

$$\text{\#:change change:window window:to to:pop-up pop-up:\#} \tag{4}$$

where the pound symbol '#' represents a sentence boundary marker, and the colon symbol ':' is used to separate the two components of each 2-tuple. Thus, the word 2-tuples are obtained by shifting one word from left to right (or, alternatively, from right to left) and grouping the next 2 consecutive words together at each shift, resulting in a sequence of overlapping pairs of words. A similar process could be used with $n = 3$ and so forth, depending on the requirements of the particular VCC application in which the present invention is embodied.

During training, the training corpus $\mathcal{T}$, may consist of a command document for the command to change a display from a pop-up to a window and another command document for the command to change a display from a window to a pop-up. The commands could include the word groupings "change pop-up to window," "change from pop-up to window," "change to window," "change window to pop-up," "change from window to pop-up," and "change to pop-up" etc. For $n = 2$, the associated word 2-tuples define the associated 2-tuple vocabulary $\overline{V}$ comprising all the word 2-tuples appearing in $\mathcal{T}$, and are summarized in Table I.

Table I

Word 2-tuples
#:change
change:pop-up
pop-up:to
to:window
change:to
change:from
from:pop-up
window:#
from:window
window:to
change:window
to:pop-up

The word n-tuple/n-tuple-document correlation matrix is then constructed in step 306 as an $\left(\overline{M} \times N\right)$ matrix $\overline{W}$ of word n-tuples by n-tuple documents, whose entries reflect, for the training corpus $\mathcal{T}$, which n-tuples appeared in what n-tuple documents. As indicated above, the entries of the word n-tuple/n-tuple-document correlation matrix $\overline{W}$ are generated by determining the number of times each of the M word n-tuples occurs in each of the N command documents. Using the above example for the commands to change a display from a pop-up to a window and vice versa, the counts of the number of

04860.P2564

- 16 -

times each of the M word n-tuples in the word n-tuple/n-tuple-document correlation

matrix $\overline{W}$ are illustrated in Table II. Counts for all other word n-tuples in the word n-

tuple/n-tuple-document correlation matrix $\overline{W}$ for the commands are zero.

<u>Table II</u>

Word n-tuple	Pop-up to Window Count	Window to pop-up Count
#:change	3	3
change:pop-up	1	0
pop-up:to	2	0
to:window	3	0
change:to	1	1
change:from	1	1
from:pop-up	1	0
window:#	3	0
from:window	0	1
window:to	0	2
change:window	0	1
to:pop-up	0	3

5

An SVD of the word n-tuple/n-tuple-document correlation matrix $\overline{W}$ having an

order of decomposition of R is then generated in step 308. The value of R can vary

depending on the values of M and N, and by balancing computational speed (associated

with lower values of R) against accuracy (associated with higher values of R). Typical

10   values for R range from 5 to 50. The SVD is generated according to the following

calculation:

$$\overline{W} \approx \hat{\overline{W}} = USV^T \tag{5}$$

where U represents the $M \times R$ matrix of left singular vectors, V^T represents the

transposition of the $N \times R$ matrix of right singular vectors, and S represents the $R \times R$

15   diagonal matrix of singular values. The generation of the left and right singular vectors,

as well as the diagonal matrix of singular values, is well-known to those skilled in the art

and thus will not be discussed further.

Given the SVD from calculation (5), a particular document d_j of $\overline{W}$ can be

determined based on the jth right singular vector v_j according to the following

5   calculation:

$$d_j = USv_j^T \tag{6}$$

Further, based on calculation (6) and using well-known mathematical functions and

properties, the value of the jth right singular vector v_j can be calculated according to the

following:

10    $$v_j = d_j^T US^{-1} \tag{7}$$

It is to be appreciated that the value US^{-1} does not change for different values of j, and

therefore the value US^{-1} can be pre-calculated in step 310, and used during classification

of input values as discussed in more detail below. This pre-calculation reduces the

computation required during command classification, thereby increasing the speed of the

15   command recognition system during operation.

Thus, from the training corpus T, a large-span semantic vector space having a

dimension R is created. Each document from the training corpus T results in a different

semantic anchor within that vector space. Furthermore, each of the semantic anchors is

based not only on the actual words of the training data but also on the order of the words

20   in the training data. The semantic anchors are used in classifying input data as a

particular command, as discussed in more detail below.

Figure 4 is a flowchart illustrating the process followed in classifying audio input

according to one embodiment of the present invention. According to one

implementation, Figure 4 illustrates the process of step 210 of Figure 2 in more detail.

As illustrated, a vector is first generated for the received sequence of word n-tuples in

step 402.

After the word agglomeration unit 105 generates the sequence of word n-tuples

5   111 for a new audio input 102, the semantic classification unit 112 creates a new

command vector d_p of order M that identifies the number of occurrences of each of the M

word n-tuples in the sequence of word n-tuples 111 in step 402. The semantic

classification unit 112 then determines the value of the right singular vector

v_p corresponding to this new sequence of word n-tuples 111 in step 404, according to the

10   following calculation (similar to calculation (7) discussed above):

$$v_p = d_p^T U S^{-1} \qquad (8)$$

As discussed above, US^{-1} is pre-computed. Therefore, the only computation which

occurs at this step is the matrix multiplication of d_p^T by the pre-computed value US^{-1}.

The result of calculation (8), v_p, is a semantic representation 116 of the new sequence of

15   word n-tuples 111 in the vector space R.

The semantic classification unit 112 then identifies one or more semantic anchors

114 to be used for comparison to the semantic representation vector v_p 116 in step 406.

According to one embodiment of the present invention, all of the semantic anchors 114

generated from the training corpus T are used for comparison, and thus all are identified

20   in step 406. Alternatively, the semantic anchors 114 can be clustered into different

groups, each group including a subset of the anchors from the training data. Any of a

variety of clustering algorithms, such as bottom-up clustering, can be used. Clustering

algorithms are well-known to those skilled in the art, and thus will not be discussed

further except as they pertain to the present invention.

The classifier then compares the semantic representation v_p 116 of the new

sequence of word n-tuples 111 to one or more of the different semantic anchors 114 in the

5   vector space previously calculated from the training corpus $\mathcal{T}$ in step 408. A "distance"

between the semantic representation vector v_p 116 and each of the one or more semantic

anchor training vectors 114 is then determined, with the "distance" representing the

similarity between the vectors. Two vectors which are similar have a small distance

between one another, whereas vectors which are less similar have larger distances

10   between one another. The semantic classification unit 112 assigns the semantic

representation vector v_p 116 to the semantic anchor training vector 114 that is closest in

distance to the semantic representation vector v_p 116 in step 410.

The distance between the semantic representation vector v_p 116 and a semantic

anchor vector v_j 114 from the training data can be calculated in a variety of different

15   manners. According to one embodiment of the present invention, the distance z between

the semantic representation vector v_p 116 and a semantic anchor vector v_j 114 from the

training data is determined by calculating the inverse cosine of the similarity between

$v_p S$ and $v_j S$ according to the following calculation:

$$z = \cos^{-1} \frac{v_p S^2 v^T_j}{\left\| v_p S \right\| \left\| v_j S \right\|} \qquad (9)$$

20   According to an alternate embodiment of the present invention, the distance z

between the semantic representation vector v_p 116 and one of the semantic anchor vectors

v_j 114 from the training data is determined by calculating the Euclidean distance between the two vectors according to the following calculation:

$$z = \sqrt{\sum_{i=1}^{R}((v_p)_i - (v_j)_i)^2} \qquad (10)$$

The comparison process in steps 408 and 410 can be accomplished in a variety of manners. According to another embodiment, comparisons for all of the identified semantic anchors 114 are made and the semantic anchor v_j 114 that is most similar to the semantic representation vector v_p 116 is selected. According to an alternate embodiment, as soon as a semantic anchor v_j 114 that is less than a threshold distance from the semantic representation vector v_p 116 is discovered, the comparison process ends and that semantic anchor v_j 114 is determined to be the "closest" in step 410.

Thus, as can be seen in Figure 4, the classification of the input command is performed using semantic inference and word agglomeration. It is a semantic representation of the word n-tuples of the audio input that is compared to the semantic anchors associated with the word n-tuples of the commands in the training data in order to classify the audio input.

Figure 5 illustrates one embodiment of a computing device suitable for use with one embodiment the present invention. According to one implementation, command recognition system 100 of Figure 1 is a hardware system 500 of Figure 5. In the illustrated embodiment of Figure 5, system 500 includes processor 502, display device 506, and input/output (I/O) devices 508, coupled to each other via a bus 510. Additionally, a memory subsystem 512, which can include one or more of cache memories, system memory (RAM), and nonvolatile storage devices (e.g., magnetic or

optical disks) is also coupled to bus 510 for storage of instructions and data for use by

processor 502. I/O devices 508 represent a broad range of input and output devices,

including keyboards, cursor control devices (e.g., a trackpad or mouse), microphones,

speakers, network or telephone communication interfaces, printers, etc. Hardware system

5   500 also includes well-known audio processing hardware and/or software to transform

analog voice data to a digital form which can be processed by system 500.

These elements 502 - 512 perform their conventional functions known in the art.

Collectively, these elements are intended to represent a broad category of hardware

systems, including but not limited to general purpose computer systems based on the

10   PowerPC® processor family of processors available from Motorola, Inc. of Schaumburg,

Illinois, or the Pentium® processor family of processors available from Intel Corporation

of Santa Clara, California.

It is to be appreciated that various components of hardware system 500 may be re-

arranged, and that certain implementations of the present invention may not require nor

15   include all of the above components. For example, a display device may not be included

in system 500. Additionally, multiple buses (e.g., a standard I/O bus and a high

performance I/O bus) may be included in system 500. Furthermore, additional

components may be included in system 500, such as additional processors (e.g., a digital

signal processor), storage devices, memories, network/communication interfaces, etc.

20   In the illustrated embodiment of Figure 5, the method and apparatus for command

recognition using semantic inference and word agglomeration according to the present

invention as discussed above is implemented as a series of software routines run by

hardware system 500 of Figure 5. These software routines comprise a plurality or series

of instructions to be executed by a processor in a hardware system, such as processor 502 of Figure 5. Initially, the series of instructions are stored on a storage device of memory subsystem 512. It is to be appreciated that the series of instructions can be stored using any conventional storage medium, such as a diskette, CD-ROM, magnetic tape, DVD,

5 ROM, Flash memory, etc. It is also to be appreciated that the series of instructions need not be stored locally, and could be received from a remote storage device, such as a server on a network, via a network/communication interface. The instructions are copied from the storage device, such as mass storage, into system memory and then accessed and executed by processor 502. In one implementation, these software routines are written in

10 the C++ programming language. It is to be appreciated, however, that these routines may be implemented in any of a wide variety of programming languages.

These software routines are illustrated in memory subsystem 512 as speech recognition instructions 520, word agglomeration instructions 521, semantic classification instructions 522, action generation instructions 523, and training

15 instructions 524. Also illustrated are analog to digital (A/D) transformation instructions 525, acoustic model(s) 526, language model(s) 527, and semantic anchors 528.

In alternate embodiments, the present invention is implemented in discrete hardware or firmware. For example, one or more application specific integrated circuits (ASICs) could be programmed with the above described functions of the present

20 invention. By way of another example, speech recognition unit 104, word agglomeration unit 105, semantic classification unit 112, and action generation unit 118 of Figure 1 could be implemented in one or more ASICs of an additional circuit board for insertion into hardware system 500 of Figure 5.

In the discussions above, the present invention is described with reference to voice recognition systems. It is to be appreciated, however, that alternate embodiments of the present invention can be used with other types of pattern recognition systems, such as visual rather than audio pattern recognition, handwriting recognition systems (e.g.,

5   optical character recognition (OCR)), etc.

It is to be appreciated that the command recognition system of the present invention can be employed in any of a wide variety of manners. By way of example, the command recognition system could be used in conventional personal computers, security systems, home entertainment or automation systems, etc.

10   Therefore, a command recognition using semantic inference and word agglomeration has been described. An audio input is provided to a speech recognizer that identifies the words in the input. The sequence of words comprising the audio input are then provided to a word agglomerator that agglomerates the words into word n-tuples. The resulting sequence of word n-tuples are then provided to a semantic classifier that

15   classifies the audio input as corresponding to a particular command. This classification process is advantageously based on a semantic representation of the word n-tuples that comprise the audio input rather than on a semantic representation of the words themselves. The appropriate action, based on the particular command associated with the semantic representation, can then be generated. The semantic representation of the word

20   n-tuples that comprise the audio input advantageously allows the present invention to accurately identify commands with different word orders or syntax.

The performance of the above system was tested in the context of the "Speakable Items" desktop command and control task defined in MacOS 9. During performance

04860.P2564   - 24 -

testing, three different setups were considered: (i) semantic inference using classical LSA with standard word-document co-occurrences, i.e., word 1-tuples (n = 1); (ii) semantic inference using extended LSA with word pair agglomeration ($n - 2$); and (iii) semantic inference using extended LSA with word triplet agglomeration ($n = 3$). The sizes of the resulting LSA matrices were 545×120, 2178×120, and 3720×120, respectively. When using R = 62 for the order of the decomposition, the average classification error rates by agglomeration size were 4.7%, 2.5%, and 1.7%, respectively, as illustrated in Table III.

Table III

Agglomeration Size	Classification Error Rate
$n = 1$	4.7 %
$n = 2$	2.5 %
$n = 3$	1.7 %

Thus, a method and apparatus for command recognition using semantic inference and word agglomeration has been described. Whereas many alterations and modifications of the present invention will be comprehended by a person skilled in the art after having read the foregoing description, it is to be understood that the particular embodiments shown and described by way of illustration are in no way intended to be considered limiting. References to details of particular embodiments are not intended to limit the scope of the claims.

CLAIMS

What is claimed is:

1 1.   A method for recognizing speech, the method comprising:

2   recognizing a sequence of words;

3   processing the sequence of words using word agglomeration; and

4   classifying the processed sequence of words as a predetermined command.

1 2.   The method of claim 1, further comprising performing an action corresponding to

2 the predetermined command.

1 3.   The method of claim 1, wherein processing comprises replacing the sequence of

2 words with an associated word n-tuple sequence.

1 4.   The method of claim 1, wherein the associated word n-tuple sequence is a

2 sequence of all strings of n consecutive words present in the sequence of words.

1 5.   The method of claim 3, wherein classifying comprises semantically inferring the

2 predetermined command from the associated word n-tuple sequence.

1 6.   The method of claim 1, wherein classifying comprises semantically inferring the

2 predetermined command from the processed sequence of words.

1 7. The method of claim 6, wherein semantically inferring comprises determining a

2 correlation between the processed sequence of words and at least one semantic anchor.

1 8. The method of claim 7, wherein the correlation is a distance between a vector

2 corresponding to the processed sequence of words and a vector corresponding to the at

3 least one semantic anchor.

1 9. The method of claim 8, wherein semantically inferring further comprises selecting

2 the predetermined command from the semantic anchor vector having the shortest

3 distance.

1 10. The method of claim 9, wherein the semantic anchor represents a one of a

2 plurality of predetermined commands.

1 11. The method of claim 7, wherein the at least one semantic anchor is derived from a

2 training data.

1 12. The method of claim 6, wherein semantically inferring the predetermined

2 command depends on the order of the words in the processed sequence of words.

1 13. The method of claim 1, wherein the classifying comprises:

2 generating a vector representation of the processed sequence of words; and

3   comparing the vector representation to a plurality of semantic anchors, wherein

4   each of the plurality of semantic anchors corresponds to one of a plurality of voice

5   commands.

1   14.   The method of claim 13, wherein the classifying further comprises:

2   selecting a semantic anchor of the plurality of semantic anchors that is most

3   similar to the vector representation; and

4   classifying the processed sequence of words as the command that corresponds to

5   the selected semantic anchor.

1   15.   The method of claim 14, wherein the selecting comprises:

2   for each of the plurality of semantic anchors, identifying the similarity between

3   the vector representation and the semantic anchor by calculating the cosine of the angle

4   between the product of the vector representation and a diagonal matrix of singular values

5   and the product of the semantic anchor and the diagonal matrix of singular values; and

6   selecting the semantic anchor of the plurality of semantic anchors that

7   corresponds to the largest cosine value as the semantic anchor that is most similar to the

8   vector representation.

1   16.   The method of claim 13, wherein the vector representation is an indication of how

2   frequently each of a plurality of word n-tuples occurs within the processed sequence of

3   words.

1   17.   The method of claim 16, wherein each of the plurality of semantic anchors is an

2   indication of how frequently each of the plurality of word n-tuples occurs with respect to

3   the corresponding command.

1   18.   The method of claim 13, wherein each of the plurality of semantic anchors

2   represents a plurality of different ways of speaking the corresponding command.

1   19.   The method of claim 13, wherein each of the plurality of semantic anchors

2   represents a plurality of different commands having the same words, but in a different

3   order.

1   20.   A machine-readable medium having stored thereon a plurality of instructions that,

2   when executed by a processor, cause the processor to recognize a voice command by:

3   recognizing a sequence of words;

4   processing the sequence of words using word agglomeration; and

5   classifying the processed sequence of words as a predetermined command.

1   21.   The machine-readable medium of claim 20, wherein the plurality of instructions

2   further cause the processor to perform an action corresponding to the predetermined

3   command.

1   22.   The machine-readable medium of claim 20, wherein the plurality of instructions

2   for processing comprises:

3   replacing the sequence of words with an associated word *n*-tuple sequence.

1   23.   The machine-readable medium of claim 22, wherein the associated word *n*-tuple

2   sequence is a sequence of all strings of *n* consecutive words present in the sequence of

3   words.

1   24.   The machine-readable medium of claim 22, wherein the instructions for

2   classifying comprises semantically inferring the predetermined command from the

3   associated word *n*-tuple sequence.

1   25.   The machine-readable medium of claim 22, wherein the instructions for

2   semantically inferring comprises determining a correlation between a semantic

3   representation of the associated word *n*-tuple sequence and at least one semantic anchor.

1   26.   The machine-readable medium of claim 25, wherein the instructions for

2   determining a correlation comprise determining a distance between a vector

3   corresponding to the semantic representation and a vector corresponding to the at least

4   one semantic anchor.

1   27.   The machine-readable medium of claim 26, wherein the instructions for

2   semantically inferring further comprises selecting the predetermined command from the

3   semantic anchor vector having the shortest distance to the a vector corresponding to the

4   semantic representation.

1   28.   The machine-readable medium of claim 25, wherein the at least one semantic

2   anchor represents a one of a plurality of predetermined commands.

1   29.   The machine-readable medium of claim 25, wherein the at least one semantic

2   anchor is derived from a training data.

1   30.   The machine-readable medium of claim 25, wherein the performance of the

2   instructions for semantically inferring the predetermined command depends on the order

3   of the words in the processed sequence of words.

1   31.   An apparatus for recognizing a voice command, the apparatus comprising:

2   a speech recognizer to recognize a sequence of words received as the voice

3   command;

4   a processor to process the sequence of words using word agglomeration; and

5   a semantic classifier, coupled to the processor, to semantically infer from a vector

6   representation of the processed sequence of words which of a plurality of predetermined

7   commands correlate to the voice command.

1   32.   The apparatus of claim 31, further comprising:

2   an action generator, coupled to the semantic classifier, to use the vector

3   representation to determine an action to be performed.

1   33.   The apparatus of claim 31, wherein the semantic classifier is further to compare

2   the vector representation to a plurality of semantic anchors, wherein each of the plurality

3   of semantic anchors corresponds to a one of the plurality of predetermined commands.

1   34.   The apparatus of claim 33, wherein the semantic classifier is further to identify a

2   semantic anchor of the plurality of semantic anchors that is most similar to the vector

3   representation, and to classify the vector representation as the one of the plurality of

4   predetermined commands that corresponds to the identified semantic anchor.

1   35.   An apparatus for recognizing a voice command, the apparatus comprising:

2   means for recognizing a sequence of words received as the voice command;

3   means for processing the sequence of words using word agglomeration; and

4   means, coupled to the means for processing, for semantically inferring from a

5   vector representation of the processed sequence of words which of a plurality of

6   predetermined commands correlate to the voice command.

1   36.   The apparatus of claim 35, further comprising means, coupled to the means for

2   semantically inferring, for using the vector representation to determine an action to be

3   performed.

1 37.   The apparatus of claim 35, wherein the means for semantically inferring further

2 comprises:

3   means for comparing the vector representation to a plurality of semantic anchors,

4 wherein each of the plurality of semantic anchors corresponds to one of a plurality of

5 predetermined commands.

1 38.   The apparatus of claim 37, wherein the means for semantically inferring further

2 comprises:

3   means for identifying a semantic anchor of the plurality of semantic anchors that

4 is most similar to the vector representation; and

5   means for classifying the vector representation as the one of the plurality of

6 predetermined commands that corresponds to the identified semantic anchor.

1 39.   A method for characterizing language comprising:

2   grouping an input sequence of words into at least one of a plurality of word n-

3 tuples, wherein each word n-tuple is a string of n consecutive words present in the

4 sequence;

5   mapping the plurality of word n-tuples into a first vector of a vector space,

6 wherein each word n-tuple is an element of the first vector; and

7   calculating a distance between the first vector and a plurality of second vectors in

8 the vector space, each of said plurality of second vectors having been previously mapped

9 from a training sequence of words.

1 40. The method of claim 39 wherein grouping the input sequence of words into at

2 least one of a plurality of word n-tuples is performed by:

3   grouping n consecutive words of the sequence into a first word n-tuple starting

4 with the left-most word in the sequence of words;

5   repeating the grouping of n consecutive words of the sequence into a subsequent

6 word n-tuple until all of the words in the sequence of words have been mapped, each

7 repetition starting with the next left-most word in the sequence of words.

1 41. The method of claim 40, wherein grouping starts with the right-most word in the

2 sequence of words, and each repetition starts with the next right-most word in the

3 sequence of words.

1 42. The method of claim 39 wherein grouping a sequence of words into at least one of

2 a plurality of word n-tuples is performed by word agglomeration.

1 43. The method of claim 39 wherein mapping the plurality of word n-tuples further

2 includes:

3   tabulating the number of times each word n-tuple occurs in the plurality of word

4 n-tuples using a word n-tuple by n-tuple document matrix, wherein entries from the

5 matrix form the first vector in the vector space.

1   44.   The method of claim 39 wherein mapping the training sequence of words further

2   includes:

3   tabulating the number of times a word n-tuple grouped from the training sequence

4   of words occurs in a set of N training documents using a word n-tuple by n-tuple

5   document matrix, wherein entries from the matrix form the plurality of second vectors in

6   the vector space, which have dimension N.

1   45.   The method of claim 39 further comprising classifying the input sequence of

2   words as representing a predetermined command corresponding to a one of the plurality

3   of second vectors closest in distance to the first vector.

ABSTRACT OF THE DISCLOSURE

A method and apparatus for command recognition using semantic inference and word agglomeration is described herein. According to one aspect of the present invention, a method for recognizing a voice command comprises recognizing a sequence of words received as the voice command. The sequence of words is further agglomerated into a sequence of word n-tuples. Semantic inference is applied to the sequence of word n-tuples to recognize the voice command.

.

SYSTEM 100 OF FIGURES

Fig. 1

```
┌─────────────────────────┐
│                         │  202
│       TRAIN SYSTEM      │
│                         │
└────────────┬────────────┘
             │
    ┌────────▼────────────────┐
    │                         │  204
    │  RECEIVE AUDIO COMMAND  │
    │         INPUT           │
    │                         │
    └────────────┬────────────┘
             │
    ┌────────▼────────────────┐
    │                         │  206
    │    PERFORM SPEECH       │
    │   RECOGNITION ON INPUT  │
    │                         │
    └────────────┬────────────┘
             │
    ┌────────▼────────────────┐
    │                         │  208
    │    PERFORM WORD         │
    │    AGGLOMERATION        │
    │      ON INPUT           │
    └────────────┬────────────┘
             │
    ┌────────▼────────────────┐
    │                         │  210
    │   PERFORM SEMANTIC      │
    │   CLASSIFICATION        │
    │     ON INPUT            │
    └────────────┬────────────┘
             │
    ┌────────▼────────────────┐
    │                         │  212
    │  USE CLASSIFIED         │
    │  INPUT TO GENERATE      │
    │     ACTION             │
    └─────────────────────────┘
```

Fig. 2

```
                    ┌────────────────────────┐
                    │  RECEIVE ONE OR MORE   │  ╱ 302
                    │ WORD SEQUENCES FOR EACH│
                    │   COMMAND DOCUMENT     │
                    └────────────────────────┘
                                │
                                │
                    ┌────────────────────────┐
                    │   AGGLOMERATE          │  ╱ 304
                    │     WORDS              │
                    │      TO                │
                    │  WORD n-TUPLES         │
                    └────────────────────────┘
                                │
                                │
        ┌──────────────────────────────────────────┐
        │          GENERATE WORD n=TUPLE/           │  ╱ 306
        │  N-TUPLE  DOCUMENT CORRELATION            │
        │                MATRIX                     │
        └──────────────────────────────────────────┘
                                │
                                ▼
                    ┌────────────────────────┐
                    │   GENERATE SINGULAR    │  ╱ 308
                    │ VALUE DECOMPOSITION OF │
                    │  WORD-DOCUMENT MATRIX  │
                    └────────────────────────┘
                                │
                                ▼
                    ┌────────────────────────┐
                    │  PRE-COMPUTE VALUES FOR│  ╱ 310
                    │    CLASSIFICATION      │
                    └────────────────────────┘
```

Fig. 3

GENERATE COMMAND
VECTOR FOR RECEIVED
SEQUENCE OF WORD N-TUPLES   402

CALCULATE RIGHT
SINGULAR VECTOR USING
COMMAND VECTOR   404

IDENTIFY ONE OR MORE
SEMANTIC ANCHORS FOR
COMPARISON   406

COMPARE RECEIVED
COMMAND TO EACH OF
THE IDENTIFIED ANCHORS   408

CHOOSE "CLOSEST"
ANCHOR AS
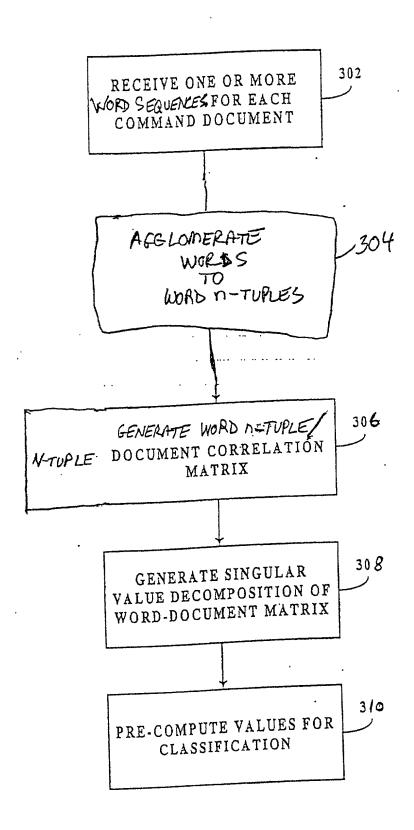CLASSIFICATION OF
COMMAND   410

Fig. 4
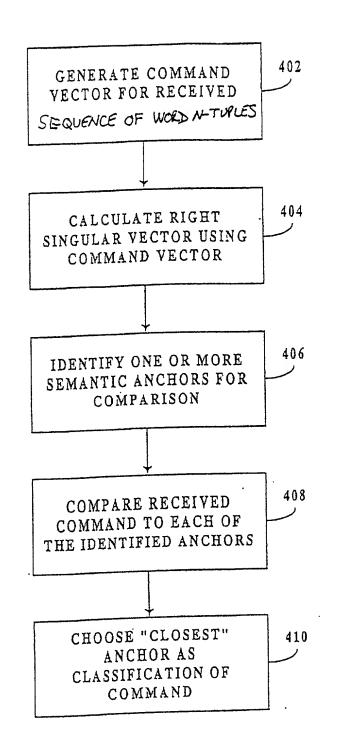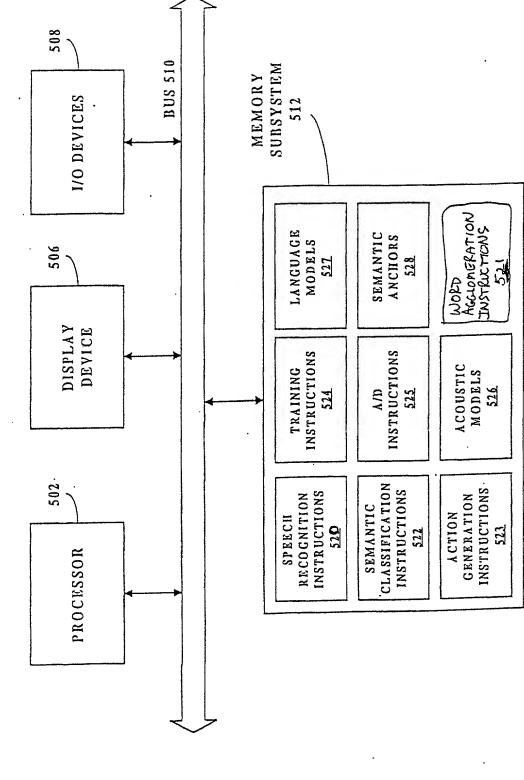
Fig. 5

Attorney's Docket No.: <u>004860.P2564</u>

DECLARATION AND POWER OF ATTORNEY FOR PATENT APPLICATION

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below, next to my name.

I believe I am the original, first, and sole inventor (if only one name is listed below) or any original, first, and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled

METHOD AND APPARATUS FOR SPEECH RECOGNITION USING SEMANTIC INFERENCE AND WORD AGGLOMERATION

the specification of which   [X]   is attached hereto.

               [ ]   was filed on _____ as ___

United States Application Number   _____

or PCT International Application Number   _____

and was amended on   _____

<p align="center">(if applicable)</p>

I hereby state that I have reviewed and understand the contents of the above-identified specification, including the claim(s), as amended by any amendment referred to above. I do not know and do not believe that the claimed invention was ever known or used in the United States of America before my invention thereof, or patented or described in any printed publication in any country before my invention thereof or more than one year prior to this application, that the same was not in public use or on sale in the United States of America more than one year prior to this application, and that the invention has not been patented or made the subject of an inventor's certificate issued before the date of this application in any country foreign to the United States of America on an application filed by me or my legal representatives or assigns more than twelve months (for a utility patent application) or six months (for a design patent application) prior to this application.

I acknowledge the duty to disclose all information known to me to be material to patentability as defined in Title 37, Code of Federal Regulations, Section 1.56.

I hereby claim foreign priority benefits under Title 35, United States Code, Section 119(a)-(d), of any foreign application(s) for patent or inventor's certificate listed below and have also identified below any foreign application for patent or inventor's certificate having a filing date before that of the application on which priority is claimed:

<u>Prior Foreign Application(s)</u>:

APPLICATION NUMBER	COUNTRY (OR INDICATE IF PCT)	DATE OF FILING (day, **month**, year)	PRIORITY CLAIMED UNDER 37 USC 119
			[ ] Yes [ ] No
			[ ] Yes [ ] No
			[ ] Yes [ ] No

I hereby claim the benefit under Title 35, United States Code, Section 119(e) of any United States provisional application(s) listed below:

APPLICATION NUMBER	FILING DATE

I hereby claim the benefit under Title 35, United States Code, Section 120 of any United States application(s) listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35, United States Code, Section 112, I acknowledge the duty to disclose all information known to me to be material to patentability as defined in Title 37, Code of Federal Regulations, Section 1.56 which became available between the filing date of the prior application and the national or PCT international filing date of this application:

APPLICATION NUMBER	FILING DATE	STATUS (ISSUED, PENDING, ABANDONED)

I hereby appoint the persons listed on Appendix A hereto (which is incorporated by reference and a part of this document) as my respective patent attorneys and patent agents, with full power of substitution and revocation, to prosecute this application and to transact all business in the Patent and Trademark Office connected herewith.

Send correspondence to:
Donna Jo Coningsby, Reg. No. 41,684, BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN, LLP
(Name of Attorney or Agent)
12400 Wilshire Boulevard, 7th Floor, Los Angeles, California 90025 and direct telephone calls to:
Donna Jo Coningsby, (503) 684-6200.
(Name of Attorney or Agent)

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Full Name of Sole/First Inventor (given name, family name)   **Jerome R. Bellegarda**

Inventor's Signature _____   Date _____

Residence _____   Citizenship _____
                (City, State)   *(Country)*

P. O. Address _____

_____

Full Name of Second/Joint Inventor (given name, family name) _____

Inventor's Signature _____   Date _____

Residence _____   Citizenship _____
                (City, State)   *(Country)*

P. O. Address _____

_____

Docket No. 004860.P2564   2

Full Name of Third/Joint Inventor (given name, family name) _____

Inventor's Signature _____   Date _____

Residence _____   Citizenship _____
_____(City , State)_____   _____(Country)_____

P. O. Address _____
_____

Full Name of Fourth/Joint Inventor (given name, family name) _____

Inventor's Signature _____   Date _____

Residence _____   Citizenship _____
_____(City , State)_____   _____(Country)_____

P. O. Address _____
_____

Full Name of Fifth/Joint Inventor (given name, family name) _____

Inventor's Signature _____   Date _____

Residence _____   Citizenship _____
_____(City , State)_____   _____(Country)_____

P. O. Address _____
_____

Full Name of Sixth/Joint Inventor (given name, family name) _____

Inventor's Signature _____   Date _____

Residence _____   Citizenship _____
_____(City , State)_____   _____(Country)_____

P. O. Address _____
_____

Full Name of Seventh/Joint Inventor (given name, family name) _____

Inventor's Signature _____   Date _____

Residence _____   Citizenship _____
_____(City, State)_____   _____(Country)_____

P. O. Address _____
_____

Full Name of Eighth/Joint Inventor (given name, family name) _____

Inventor's Signature _____   Date _____

Residence _____   Citizenship _____
_____(City, State)_____   _____(Country)_____

P. O. Address _____
_____

Full Name of Ninth/Joint Inventor (given name, family name) _____

Inventor's Signature _____   Date _____

Residence _____   Citizenship _____
_____(City, State)_____   _____(Country)_____

P. O. Address _____
_____

Full Name of Tenth/Joint Inventor (given name, family name) _____

Inventor's Signature _____   Date _____

Residence _____   Citizenship _____
_____(City, State)_____   _____(Country)_____

P. O. Address _____
_____

Appendix A

I hereby appoint BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP, a firm including: William E. Alford, Reg. No. 37,764; Farzad E. Amini, Reg. No. 42,261; William Thomas Babbitt, Reg. No. 39,591; Carol F. Barry, Reg. No. 41,600; Jordan Michael Becker, Reg. No. 39,602; Lisa N. Benado, Reg. No. 39,995; Bradley J. Bereznak, Reg. No. 33,474; Michael A. Bernadicou, Reg. No. 35,934; Roger W. Blakely, Jr., Reg. No. 25,831; R. Alan Burnett, Reg. No. 46,149; Gregory D. Caldwell, Reg. No. 39,926; Andrew C. Chen, Reg. No. 43,544; Thomas M. Coester, Reg. No. 39,637; Donna Jo Coningsby, Reg. No. 41,684; Florin Corie, Reg. No. 46,244; Dennis M. deGuzman, Reg. No. 41,702; Stephen M. De Klerk, Reg. No. P46,503; Michael Anthony DeSanctis, Reg. No. 39,957; Daniel M. De Vos, Reg. No. 37,813; Sanjeet Dutta, Reg. No. P46,145; Matthew C. Fagan, Reg. No. 37,542; Tarek N. Fahmi, Reg. No. 41,402; George Fountain, Reg. No. 37,374; James Y. Go, Reg. No. 40,621; James A. Henry, Reg. No. 41,064; Willmore F. Holbrow III, Reg. No. P41,845; Sheryl Sue Holloway, Reg. No. 37,850; George W Hoover II, Reg. No. 32,992; Eric S. Hyman, Reg. No. 30,139; William W. Kidd, Reg. No. 31,772; Sang Hui Kim, Reg. No. 40,450; Walter T. Kim, Reg. No. 42,731; Eric T. King, Reg. No. 44,188; Erica W. Kuo, Reg. No. 42,775; George B. Leavell, Reg. No. 45,436; Gordon R. Lindeen III, Reg. No. 33,192; Jan Carol Little, Reg. No. 41,181; Kurt P. Leyendecker, Reg. No. 42,799; Joseph Lutz, Reg. No. 43,765; Michael J. Mallie, Reg. No. 36,591; Andre L. Marais, under 37 C.F.R. § 10.9(b); Paul A. Mendonsa, Reg. No. 42,879; Clive D. Menezes, Reg. No. 45,493; Chun M. Ng, Reg. No. 36,878; Thien T. Nguyen, Reg. No. 43,835; Thinh V. Nguyen, Reg. No. 42,034; Dennis A. Nicholls, Reg. No. 42,036; Daniel E. Ovanezian, Reg. No. 41,236; Kenneth B. Paley, Reg. No. 38,989; Gregg A. Peacock, Reg. No. 45,001; Marina Portnova, Reg. No. P45,750; William F. Ryann, Reg. 44,313; James H. Salter, Reg. No. 35,668; William W. Schaal, Reg. No. 39,018; James C. Scheller, Reg. No. 31,195; Jeffrey Sam Smith, Reg. No. 39,377; Maria McCormack Sobrino, Reg. No. 31,639; Stanley W. Sokoloff, Reg. No. 25,128; Judith A. Szepesi, Reg. No. 39,393; Vincent P. Tassinari, Reg. No. 42,179; Edwin H. Taylor, Reg. No. 25,129; John F. Travis, Reg. No. 43,203; Joseph A. Twarowski, Reg. No. 42,191; Thomas A. Van Zandt, Reg. No. 43,219; Lester J. Vincent, Reg. No. 31,460; Glenn E. Von Tersch, Reg. No. 41,364; John Patrick Ward, Reg. No. 40,216; Mark L. Watson, Reg. No. P46,322; Thomas C. Webster, Reg. No. P46,154; and Norman Zafman, Reg. No. 26,250; my patent attorneys, with offices located at 12400 Wilshire Boulevard, 7th Floor, Los Angeles, California 90025, telephone (503) 684-6200, and Justin M. Dillon, Reg. No. 42,486 and Raul Martinez, Reg. No. 46,904, and Dov Rosenfeld, Reg. No. 38,687, my patent agents; and James R. Thein, Reg. No. 31,710; with full power of substitution and revocation, to prosecute this application and to transact all business in the Patent and Trademark Office connected herewith.